

TABLE OF CONTENTS

1.	Identification Page.....	1
2.	Table of Contents	2
3.	Real Party in Interest	3
4.	Related Appeals and Interferences	4
5.	Status of Claims	5
6.	Status of Amendments	6
7.	Summary of Claimed Subject Matter	7
8.	Grounds of Rejection to be Reviewed on Appeal	9
9.	Arguments	10
10.	Conclusion	15
11.	Claims Appendix	16
12.	Evidence Appendix	21
13.	Related Proceedings Appendix	22

Real Party in Interest

The present application has been assigned to International Business Machines Corporation, Armonk, New York.

Related Appeals and Interferences

Applicant asserts that no other appeals or interferences are known to the Applicant, the Applicant's legal representative, or assignee which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

Status of Claims

Claims 1-2 and 4-30 are pending in the application. Claims 1-30 were originally presented in the application. Claim 3 has been canceled without prejudice. Claims 1-2 and 4-30 stand finally rejected as discussed below. The final rejections of claims 1-2 and 4-30 are appealed. The pending claims are shown in the attached Claims Appendix.

Status of Amendments

All claim amendments have been entered by the Examiner, including amendments to the claims proposed after the final rejection.

Summary of Claimed Subject Matter

The present invention relates to a method of managing transactions processed through a plurality of computers communicably linked in a business transaction environment, such as business to business, business to client or business to government. See, e.g., pp. 5-6, paragraph 0016.

One embodiment, e.g., claim 1, provides a method of maintaining a database for managing of a plurality of transactions through two or more applications, e.g., elements 31, 32, 41, 42, 51 and 52 of Figure 1, in a business transaction environment, e.g., element 100 of Figure 1. See pp. 7-8, paragraph 0021. Each application has at least one associated log file, e.g., log file 33 for application 31 and log file 45 for application 41 in Figure 1. Each transaction is defined by one or more steps configured to complete the transaction. See Figure 3A and pp. 12-13, paragraph 0032. The method comprises accessing each of the respective associated log files wherein at least two of the associated log files are of different formats. See Figure 1, pp. 7-8, paragraph 0021. The method further comprises performing a process, e.g., process 200 shown in Figure 2, for each new log entry recorded (See pp. 9-10, paragraphs 0024 and 0025) in the respective associated log file being accessed. The process comprises determining whether the new log entry comprises one or more required filed, e.g. transaction ID, using mapping rules, e.g., element 49 of Figure 1, see p. 10, paragraph 0028 and steps 230/240 of Figure 2A, extracting information from the new log entry only if the new log entry comprises the one or more required fields, see step 260 of Figure 2A and pp. 10-11, paragraph 0029, and storing the extracted information as a plurality of transaction record to a database, see step 310 of Figure 2B and p. 11, paragraph 0030.

Another embodiment, e.g., claim 21, provides a computer-readable storage medium containing a program which, when executed, performs an operation of maintaining a database for managing of a plurality of transactions through two or more applications, e.g., elements 31, 32, 41, 42, 51 and 52 of Figure 1, in a business transaction environment, e.g., element 100 of Figure 1. See pp. 7-8, paragraph 0021.

Each application has at least one associated log file, e.g., log file 33 for application 31 and log file 45 for application 41 in Figure 1. Each transaction is defined by one or more steps configured to complete the transaction. See Figure 3A and p. 12-13, paragraph 0032. The method comprises accessing each of the respective associated log files wherein at least two of the associated log files are of different formats, See Figure 1, pp. 7-8, paragraph 21. The method further comprises performing a process, e.g., process 200 in Figure 2, for each new log entry recorded (See pp. 9-10, paragraphs 0024 and 0025) in the respective associated log file being accessed. The process comprises determining whether the new log entry comprises one or more required fields using mapping rules, e.g., element 49 of Figure 1, see p. 10, paragraph 0028 and steps 230/240 of Figure 2A, extracting information from the new log entry only if the new log entry comprises the one or more required fields, see step 260 of Figure 2A and pp. 10-11, paragraph 0029, and storing the extracted information as a plurality of transaction record to a database, see step 310 of Figure 2B and p. 11, paragraph 0030.

Yet another embodiment, e.g., claim 29, provides a computer, e.g., element 40 of Figure 1. The computer comprises a database maintenance program, e.g., element 44 of Figure 1. The database maintenance program is configured for managing a process of a plurality of transactions through two or more applications, e.g., elements 31, 32, 41, 42, 51 and 52 of Figure 1, in a business transaction environment, e.g., element 100 of Figure 1. Each application has at least one associated log file, e.g., log file 33 for application 31 and log file 45 for application 41 in Figure 1. Each transaction is defined by one or more steps configured to complete the transaction. See Figure 3A and pp. 12-13, paragraph 0032. For each new log entry recorded in the log files, the program determines whether the new log entry comprises one or more required filed using mapping rules, e.g., element 49 of Figure 1, see p. 10, paragraph 0028 and steps 230/240 of Figure 2A, extracts information from the new log entry only if the new log entry comprises the one or more required fields, see step 260 of Figure 2A and pp. 10-11, paragraph 0029, and stores the extracted information as a plurality of transaction record to a database, see step 310 of Figure 2B and p. 11, paragraph 0030.

Grounds of Rejection to be Reviewed on Appeal

1. Claims 1-2 and 4-30 stand rejected under 35 U.S.C.103(a) as being unpatentable over *Matson et al.* (U.S. Patent 6,668,254, hereinafter *Matson*) in view of *Landry* (U.S. Patent 5,649,117).

ARGUMENTS

Obviousness of Claims 1-2 and 4-30 over *Matson* in view of *Landry*.

Prosecution History

Following withdrawal of the finality of a previous rejection, the Examiner issued a new final rejection on March 21, 2006. In that final office action, claims 1-2, and 4-30 were rejected under 35 U.S.C. § 103(a) as being unpatentable over *Matson* (U.S. Patent 6,666,254) in view of *Laundry* (U. S. Patent 5,649,117). Applicants appeal this rejection.

The Applicable Law

The Examiner bears the initial burden of establishing a *prima facie* case of obviousness. See MPEP § 2142. To establish a *prima facie* case of obviousness three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one ordinary skill in the art to modify the reference or to combine the reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. See MPEP § 2143. The present rejection fails to establish at least the third criterion.

The References

Matson discloses an import manager for importing product data to a product database from different sources and in different formats (Abstract). *Matson* teaches downloading a dataset from a data source, reviewing the downloaded dataset using a differential analysis, extracting the downloaded dataset to a standard format, and storing the extracted dataset to the product database (Figure 2, column 4 lines 34-49, column 5 lines 18-19, and column 6 line 66 - column 7 line 2). *Matson* teaches using a differential analysis to determine if the current downloaded dataset includes new data,

and parsing (extracting) every field from the download dataset to a standard format (column 5 lines 27-29, and column 5 lines 35-39). *Matson* specifically teaches extracting all information from a dataset, regardless of whether any particular fields are present (column 5 lines 27-29, and column 5 lines 35-39).

Landry teaches a system and method for paying bills without requiring interaction with payors (Abstract).

Applicants' Arguments

First Argument:

Applicants submit that *Matson* and *Landry*, alone or in combination, do not teach, show or suggest all the limitations recited in claims 1, 21 and 29, and claims dependent thereon. For example, the combination of *Matson* and *Landry* does not teach determining whether a new log entry comprises one or more required fields, as a distinct step from, and a condition of, data extraction.

The Examiner argues that *Matson* teaches determining whether the new log entry comprises one of more required fields using mapping rules that describe a location and format of at least the one or more required fields in the respective associated log files at Col. 3 lines 9-16, Col. 4 lines 16-21, Col. 5 lines 16-21, Col. 5 lines 22-38. Respectfully, *Matson* does not teach, show or suggest determining whether a new log entry includes one of more required fields, as claimed. A careful review of the portions of *Matson* cited by the Examiner show an absence of any teaching of determining whether a new log entry includes one of more required fields. Rather, *Matson* teaches conducting a differential analysis to the whole dataset and extracting all the information from the dataset (column 4 lines 23-26, column 4 lines 36-38, column 5 lines 27-29, and column 5 lines 35-39).

The Examiner nevertheless asserts that the step of determining whether a new log entry includes one of more required fields is implicit in the parsing performed by *Matson*. Specifically, the Examiner asserts that *Matson* teaches determining if a new log entry comprises one or more required fields because *Matson* teaches "Parsing may be defined as extracting information from the supplier-specific data format" and "In

particular, the following fields in a supplier data record should be parsed (or constructed): supplier name, supplier product number, manufacturer name, manufacturer product number, vender name, and vender product number". Examiner's Advisory Action, p. 2; mailed June 9, 2006.

Applicants submit that, by parsing, *Matson* teaches extracting information from a supplier data file as completely as the supplier format allows and converting the extracted information to an XML file (column 5 lines 22-39). Therefore, *Matson* teaches extracting fields by parsing, but *Matson* does not require an independent and prior step of determining whether the supplier data file comprises one or more required fields.

Therefore, *Matson* does not teach determining whether a new log entry comprises required fields, as set forth in the claims. Therefore, the rejection is believed to be improper, and Applicants respectfully request that the rejection be withdrawn and the claims be allowed.

Second Argument:

Relatedly, *Matson* does not teach, show or suggest extracting information from the new log entry only if the new log entry comprises the one of more required fields. The Examiner asserts that *Matson* teaches extracting information from the new log entry only if the new log entry comprises the one of more required fields at col. 6 lines 14-16. In fact, the cited paragraph is directed to discarding extracted information after a differential analysis. Further, the extracting taught by *Matson* is non-selective. That is, *Matson* teaches conducting a differential analysis to the whole dataset and extracting all the information from the dataset (column 4 lines 23-26, column 4 lines 36-38, column 5 lines 27-29, and column 5 lines 35-39). Thus, to the contrary of the Examiner's assertion, *Matson* specifically teaches extracting all information from a dataset, regardless of whether any particular fields are present (column 5 lines 27-29, and column 5 lines 35-39). Thus, no determination of whether required fields are present in the dataset is done by *Matson*, and the data extraction of *Matson* is not conditioned on the presence of *required* fields.

Therefore, *Matson* does not teach conditioning extracting a new log entry on a prior determination of whether the new log entry comprises required fields, as set forth

in the claims. Therefore, the rejection is believed to be improper, and Applicants respectfully request that the rejection be withdrawn and the claims be allowed.

Third Argument:

Furthermore, *Matson* does not teach using mapping rules that describe a location and format of one or more required fields to determine whether a new log entry comprises the one or more required fields, as set forth in the claims. This conclusion follows because, as noted above, no determination of whether required fields are present in the dataset is done by *Matson*. Since *Matson* does not examine the dataset for required fields, it follows that *Matson* does not teach using mapping rules that describe a location and format of one or more required fields to determine the presence of one or more required fields.

The Examiner further asserts that *Matson* teaches using mapping rules that describe a location and format of one or more required fields because *Matson* teaches parsing “which is a mapping rules, [sic] [and] parsing inherently include [sic] location and format.” Examiner’s Advisory Action, p. 2; mailed June 9, 2006.

First, Applicants submit that *Matson* does not define parsing as a mapping rule. Parsing is generally understood as a process of analyzing a continuous stream of input, for example read from a file or a keyboard, in order to determine its grammatical structure. See “<http://en.wikipedia.org/wiki/Parsing>”. Therefore, *Matson* does not teach parsing as a mapping rule which describes a location and format of at least one or more required fields. Second, to the extent an argument on the basis of inherency can be made in a rejection under 35 USC 35 U.S.C. § 103(a), such an argument requires that the suggested inherent aspect necessarily be present. MPEP §2112 (IV). “In relying upon the theory of inherency, the examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art.” *Ex parte Levy*, 17 USPQ2d 1461, 1464 (Bd. Pat. App. & Inter. 1990) (emphasis in original). In this case, the Examiner provides no support that the suggested inherent aspect of parsing is necessarily present. Further, Applicants have provided one definition (above) that is devoid of such a requirement. Therefore, the rejection is improper and

Applicants respectfully request that the rejection be withdrawn and the claims be allowed.

Fourth Argument:

Landry teaches a system and method for paying bills without requiring interaction with payors. However, because the rejection is believed to be overcome with respect to *Matson* for the reasons given above, the rejection based on the combination of *Landry* and *Matson* is also believed to be overcome.

Summary of Arguments:

Accordingly, the references, *Matson* and *Landry*, alone or in combination, do not teach, show or suggest determining whether a new log entry comprises one or more required fields using mapping rules that describe a location and a format of at least the one or more required fields, and extracting information from the new log entry only if the new log entry comprises the one or more required fields, as recited in claims 1, 21 and 29, and claims dependent thereon.

CONCLUSION

The Examiner errs in finding that claims 1-2 and 4-30 are unpatentable over *Matson et al.* (U.S. Patent 6,668,254, hereinafter *Matson*) in view of *Landry* under 35 U.S.C. § 103(a). Withdrawal of the rejection and allowance of all claims is respectfully requested.

Respectfully submitted, and
S-signed pursuant to 37 CFR 1.4,

/Gero G. McClellan, Reg. No. 44,227/
Gero G. McClellan
Registration No. 44,227
Patterson & Sheridan, L.L.P.
3040 Post Oak Blvd. Suite 1500
Houston, TX 77056
Telephone: (713) 623-4844
Facsimile: (713) 623-4846
Attorney for Appellant(s)

CLAIMS APPENDIX

1. (Previously Presented) A method of maintaining a database for managing a process of a plurality of transactions through two or more applications in a business transaction environment, each application having at least one associated log file, and each transaction being defined by one or more steps configured to complete the transaction, the method comprising:

accessing each of the respective associated log files, wherein at least two of the associated log files are of different formats;

for each new log entry recorded in the respective associated log file being accessed:

(i) determining whether the new log entry comprises one or more required fields using mapping rules that describe a location and format of at least the one or more required fields in the respective associated log file;

(ii) extracting information from the new log entry only if the new log entry comprises the one or more required fields; and

(iii) storing the information as a plurality of transaction records to a database.

2. (Previously Presented) The method of claim 1, further comprising receiving a notification message from the respective associated log file indicating that the new log entry has been recorded in the respective associated log file.

3. (Cancelled)

4. (Previously Presented) The method of claim 1, wherein the information is extracted from the new log entry using the mapping rules providing the format and the location of the information in the new log entry.

5. (Previously Presented) The method of claim 1, further comprising determining whether the plurality of transaction records satisfies an undesirable condition; and

executing an action responsive to the undesirable condition if the plurality of transactions satisfies the condition.

6. (Original) The method of claim 5, wherein the condition is whether a number of the plurality of transaction records indicative of active transactions exceeds a predefined numerical limit.

7. (Original) The method of claim 5, wherein the condition is whether any of the plurality of transaction records indicative of active transactions has a time duration exceeding a predefined time limit.

8. (Original) The method of claim 5, wherein executing the action comprises sending a notification message alerting the condition.

9. (Original) The method of claim 5, wherein the action comprises executing a computer program for resolving the condition.

10. (Original) The method of claim 1, wherein the one or more required fields comprises at least one of a transaction identifier, a step identifier, and a time stamp.

11. (Original) The method of claim 10, wherein step identifier is a unique identifier associated with a step of the transaction.

12. (Original) The method of claim 10, wherein the time stamp indicates a time at which the step started.

13. (Previously Presented) The method of claim 1, wherein the information comprises at least one of a transaction type, a transaction origin, and a transaction destination, the transaction type, the transaction origin and the transaction destination identifying the transaction record.

14. (Original) The method of claim 13, wherein the transaction type describes the type of transaction.
15. (Original) The method of claim 13, wherein the transaction origin describes an entity that originated the transaction.
16. (Original) The method of claim 13, wherein the transaction destination describes a final destination of the transaction.
17. (Original) The method of claim 1, wherein storing the information comprises storing the information to the database as one of a transaction record and a step record, the transaction record being defined by one or more step records.
18. (Original) The method of claim 17, wherein the information comprises at least one of a step type and a step location, the step type and the step location identifying the step record.
19. (Original) The method of claim 18, wherein the step type describes the operation performed by one of the two or more applications at the time the new log entry is recorded.
20. (Original) The method of claim 18, wherein the step location describes a computer of at least one of the two or more applications.
21. (Previously Presented) A computer-readable storage medium containing a program which, when executed by a processor, performs an operation of maintaining a database for managing a process of a plurality of transactions through two or more applications in a business transaction environment, each application having at least one associated log file, each transaction being defined by one or more steps configured to complete the transaction, the operation comprising:

accessing each of the respective associated log files, wherein at least two of the associated log files are of different formats;

for each new log entry recorded in the respective associated log file being accessed:

(i) determining whether the new log entry comprises one or more required fields using mapping rules that describe a location and format of at least the one or more required fields in the respective associated log file;

(ii) extracting information from the new log entry only if the new log entry comprises the one or more required fields; and

(iii) storing the information as a plurality of transaction records to the database.

22. (Previously Presented) The computer-readable storage medium of claim 21, further comprising:

determining whether the plurality of transaction records satisfies a condition; and
executing an action if the plurality of transactions satisfies the condition.

23. (Previously Presented) The computer-readable storage medium of claim 21, wherein the one or more required fields comprises at least one of a transaction identifier, a step identifier, and a time stamp.

24. (Previously Presented) The computer-readable storage medium of claim 21, wherein storing the information as a plurality of transaction records comprises storing the information to the database as one of a transaction record and a step record, the transaction record being defined by one or more step records.

25. (Previously Presented) The computer-readable storage medium of claim 21, wherein the information is extracted from the new log entry using the set of mapping rules providing the format and the location of the information in the new log entry.

26. (Previously Presented) The computer-readable storage medium of claim 22, wherein the condition is whether a number of the plurality of transaction records indicative of active transactions exceeds a predefined limit.

27. (Previously Presented) The computer-readable storage medium of claim 22, wherein the condition is whether any of the plurality of transaction records indicative of active transactions has a time duration exceeding a predefined time limit

28. (Previously Presented) The computer-readable storage medium of claim 22, wherein executing the action comprises sending a notification message alerting the condition.

29. (Previously Presented) A computer, comprising:
a database maintenance program for managing a process of a plurality of transactions through two or more applications in a business transaction environment, each application having at least one associated log file, wherein at least two of the associated files are of different format and wherein each transaction is defined by one or more steps configured to complete the transaction; and
for each new log entry recorded in the associated log files, the transaction management program, when executed, performs an operation comprising:
determining whether the new log entry comprises one or more required fields using mapping rules that describe a location and format of at least the one or more required fields in the respective associated log file;
extracting information from the new log entry only if the new log entry comprises the one or more required fields; and
storing the information as a plurality of transaction records to the database.

30. (Previously Presented) The computer of claim 29, further comprising:
determining whether the plurality of transaction records satisfies a condition; and
executing an action if the plurality of transactions satisfies the condition.

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

None.